famCNV v2.0

1 Background

The two main methods for genome-wide Copy Number Variation (CNV) identification are based on the analysis of intensity signals from clone-based comparative genomic hybridization (array CGH) and from SNP genotyping array.
The intensity measures used to identify regions with different copy number are the normalized probe signal intensities, typically $\log_2$ ratios of intensity for aCGH data, and the $\log R$ Ratio from genotyping arrays.

For aCGH platforms DNA from a test sample and from the reference sample labeled with fluorophores are hybridized to a microarray including thousands of DNA probes complementary to targeted genomic regions. The relative fluorescence intensities of the test DNA versus the reference DNA, transformed as $\log_2$ ratios, is the metric used for assessing relative copy number.

In the Illumina SNP genotyping assay two probes are used to detect the presence of the two different alleles at each SNP. The alleles measured by the X channel (Cy5 dye) are called the A alleles, whereas the alleles measured by the Y channel (Cy3 dye) are called the B alleles. Two values are automatically generated from Illumina BeadStudio software from the X and Y values, the $\log RRatio$ and the B allele frequency. The $\log RRatio$ (LRR) is a measure proportional to the total signal intensity (and therefore the copy number). The B Allele Frequency (BAF) is a measure of the relative contribution to the total signal from the B allele (and therefore reflects the allelic composition).
Without getting into the details of the BeadStudio calculations, a simplified way to understand the derivation of LRR and BAF measurements from X/Y intensity measurements is:

$$\begin{aligned} LRR &= \log_2(X+Y) \\ BAF &= \frac{Y}{X+Y} \end{aligned}$$

LRR and BAF can be also generated from the Affymetrix SNP genotyping arrays, for example using PennCNV-Affy to read Affymetrix raw CEL files where the results of intensity calculations are stored and generate LRR and BAF values (http://www.openbioinformatics.org/penncnv/penncnv_tutorial_affy_gw6.html).

famCNV uses an intensity measure proxy for the number of copies such as the $\log_2$ ratios from aCGH or the LRR from genotyping arrays to assess association of the raw intensity data with a quantitative trait using family data.
The method does not make copy number calls at all, nor does it attempt to define the nature of any copy number variation. famCNV uses “raw” intensity data to avoid the information loss and loss of power (and introduction of errors) that occurs when translating intensity signals into discrete copy number calls.

When dealing with related samples it is necessary to model the phenotypic resemblance among family member that could be determined by polygenic effects. Therefore for family data the association between the intensity signal and the quantitative trait is calculated using a mixed model where the polygenic relationship between individuals is modeled as a random component (additional random components can be potentially added using a mixed model such as a common familial environment).

In the mixed model the data are modeled as:

$$y_i = \mu + \Sigma_j \beta_j f_{ij} + g_i + e_i$$

where y is the phenotype of the ith individual, μ the grand trait mean, f_{ij} the value of the fixed effect and β the estimate of the fjth fixed effect, g_i and e_i are the random polygenic and environmental effects, respectively.

The phenotypic variance/covariance matrix of the individuals may be written as:

$$\omega = 2\Phi\sigma_g^2 + I\sigma_e^2$$

where 2Φ is the matrix of the expected proportion of alleles shared IBD over the genome between each pair of subjects, I is an identity matrix, and σ_g^2 and σ_e^2 the polygenic and environmental variance, respectively.
The random effects are assumed to follow a multivariate normal distribution with zero mean and numerical procedures can be used to estimate the variance component parameters σ_g^2 and σ_e^2 and the likelihood of the data.

A likelihood ratio test between a model where the intensity signal is modeled in the fixed effect $L(\mu, \sigma_g^2, \sigma_e^2, \beta | y, INT)$ – where INT is a matrix of the intensity signals – and a model where its effects are assumed to be equal to 0, namely $L(\mu, \sigma_g^2, \sigma_e^2, \beta = 0 | y, INT)$ can be than used to asses the significance of the association between the intensity signal and the quantitative trait.

Note: In famCNV results can be derived including BAF as a fixed effect in both the null and the full models, to account for possibly unequal allele A and allele B signals; indeed in this case, an LRR signal might occur due to genotype-specific differences in the absence of a true copy number effect.

2 Running famCNV

famCNV is a command-line java jar executable: open up a command prompt or terminal window in the location where you have saved famCNV.jar and perform all analyses by typing commands as described below:

```
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap
```

where we expect four files: in this case:

```
mydata.ped
mydata.trmap
mydata.int
mydata.mrkmap
```

famCNV detects the number of available processors and run in a multithreadeding mode. If you want to limit the number of threads you should use the option `--thrads number`.

3 Input file format

famCNV expects four input files with predefined formats. All files follow more or less the PLINK format. Users are advised to check the validity of the files before running the software. All files can be space or tab delimited.

3.1 PED file

This file contains the pedigree information.
The first six columns are mandatory:

```
Family ID
Individual ID
Paternal ID
Maternal ID
Sex <1=male; 2=female; 0=unknown>
Phenotype
Twin's zygosity <DZ=dizygotic; MZ=monozygotic; 0=other>
```

The IDs are *numeric*: the combination of family and individual ID should uniquely identify a person. A PED file must have one and only one phenotype in the sixth column. The phenotype can only be an affection status column. Affection status column is not taken into account in the analyses. The rest of the columns contains the traits that will be tested.

Note: For subjects whose parent information are not available mock parents could be added in the PED file. Such mock parents should have `Paternal ID` and `Maternal ID` set to `0` and traits set to missing values. It is not necessary to add information about these mock parents neither in the INT file nor in the covariate file. If mock parents are not available a *beta* functionality of famCNV will try to add them.
Note: Missing values are represented by any of the following: "`X`", "`x`", "`NA`".

Hint: The remaining PED columns should refer to the rows of the TRMAP file (see below). Hence, columns of the PED file should be in the same order of rows in the TRMAP file.
Hint: This file does not have a header row.

3.2 TRMAP file

This file contains the trait info.
It has as many rows as the quantitative traits that will be tested and consists of four columns:

```
TRAIT
CHROM
BP_FROM
BP_TO
```

The row order of the transcripts follows the column order of the transcriptional levels in the PED file. The last three columns are specific for the analysis of transcript data and allow the user to limit the analysis on those markers located within a user-defined window overlapping the transcripts position (see also the `--window` option, Section 6).

Hint: If the `--window` option will not be used (set to false) with gene expression data, or when analyzing other kind of quantitative traits, the PED columns would contain just the trait's values and the TRMAP file would contain only names under the TRAIT column and the rest of column would have a missing value.
Hint: This file does not have a header row.

3.3 INT file

This file contains the intensity data, such as $\log_2$ ratios from comparative genomic hybridization (CGH) or BeadStudio LRR. Optionally, the BAF can be also added in the file. Every row corresponds to an individual. The first 2 columns are mandatory, the remaining columns correspond to the signal intensity data:

```
Family ID
Individual ID
Int_1
Int_2
...
Int_N
```

If the data contains both LRR and BAF they should come in pairs, with the BAF first:

```
Family ID
Individual ID
Ball_1
Int_1
Ball_2
Int_2
...
Ball_N
Int_N
```

So, the first LRR and BAF would occupy columns 3 and 4 and so forth.

Note: Missing values are represented by any of the following: "`X`", "`x`", "`NA`".

Hint: INT columns should refer to the rows of the MRKMAP file (see below). Hence, columns of the INT file should be in the same order of rows in the MRKMAP file.
Hint: In order to correctly use a file containing both LRR and BAF the `--Ball true` option should be used.
Hint: This file does not have a header row.

3.4 MRKMAP file

This file contains the map information for the intensity signals.
Every row corresponds to a signal and the order follows the same order of the columns in the INT file. It has 3 columns, the first 2 being mandatory:

```
NAME
CHROM
POSITION (BP)
```

Hint: This file does not have a header row.

4 Covariate file

famCNV supports the inclusion of one or more covariates in the analysis. Covariates are given in a separate file and can be included in the analysis using the option:

```
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap --cov cov.txt
```

The covariate file should be formatted in a similar manner to the phenotype file. If an individual is not present in the covariate file, or if the individual has a missing phenotype value for the covariate, then that individual will be excluded from the association analysis.
It has 3 mandatory columns:

```
Family ID
Individual ID
COVARIATE_1
```

Additional covariates can be provided in the following columns, e.g.:

```
Family ID
Individual ID
AGE
SMOKE
BMI
```

famCNV (in its current version) will include all covariates that are present in the covariates file. So, if the user wants to perform 3 types of analyses, one including only AGE as covariate and one including AGE and BMI, then he should create two covariate files, `cov1.txt`:

```
Family ID
Individual ID
AGE
```

and `cov2.txt`:

```
Family ID
Individual ID
AGE
BMI
```

and run

```
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap --cov cov1.txt
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap --cov cov2.txt
```

Note: Missing values are represented by any of the following: "`X`", "`x`", "`NA`".

Hint: This file does not have a header row.

5 Output

famCNV produces a tab-delimited output. It has 12 columns:

```
TRAIT
CHROM
BP_FROM
BP_TO
MARKER
POSITION (BP)
Lnlk_Null
Lnlk_Full
No_df_Null
No_df_Full
CHI^2
P
```

Note: When using intensity measures from SNP data a number of consecutive probes might show association with the same quantitative trait. To understand whether they are indicating the presence of the same CNV, the LRR data from all probes in the region might be averaged and then the mean used as an additional “marker”. If the individual associations reflect the action of a single CNV spanning several probes, combining them should improve the signal-to-noise ratio.

Hint: To redirect the output to a text file use the option `--output file[ath]`.
Hint: To create an output without header line use the option `--header false`.
Hint: To evaluate the contribution of the sample to the final statistics use the option `--relc threshold` (see Section 6 for details)
Hint: To print variances use the option `--variance true` (see Section 6 for details)

6 Options list

`--help`	Print a help message and exit
`--ped filepath`	PED file – *mandatory*
`--trt filepath`	file containing the trait information – *mandatory*
`--ints filepath`	file containing the intensity data – *mandatory*
`--mrk filepath`	file containing the map information for the intensity signals – *mandatory*
`--cov filepath`	file containing the covariates – *optional*
`--output filepath`	file where the output is write – *optional, default: standard output*
`--header <true|false>`	whether the output has a header – *optional, default: true*
`--inverse <true|false>`	whether traits should be transformed to them corresponding quantile in a standard normal transformation – *optional, default false*

`--Ball <true|false>` whether B allele frequencies are available. Note that when the dataset contains both LRR and BAF, the command line must contain the parameter `--Ball true` – *optional, default false*

`--window bp` window overlapping the trait location. When analyzing gene expression data it is possible to define a window overlapping the transcript location by using the option `--window bp` so that only the markers falling within the specified number of base pairs upstream and downstream of the transcript (as defined in the MRKMAP file) will be included in the analyses – *optional, default: no window considered*

`--relc threshold` whether the contribution of the sample to the final statistics must be evaluated. It allows one to verify whether the positive signal has been generated by a uniform contribution of the families within the sample or by a strong contribution of a small number of families. This information would help distinguishing between putative common/rare CNVs. This option will generate two additional columns, one reporting the percentage of families showing a positive contribution and the second one the Gini coefficient assessed on their contribution to the chi-square statistics – *optional, default: false*

`--variance <true|false>` whether the variance is printed – *optional, default: false*

`--verbose <true|false>` whether verbose – *optional, default: false*

`--threads number` the number of threads to use – *optional, default: all available*

7 Example

Example files can be found inside the data folder. `mydata.ped` contains data for 644 individuals organized in 149 families; it also contains transcriptional levels for 3 transcripts, which are described in the `mydata.trmap` file. The LRR and BAF values for 15 SNPs are given in `mydata.int` and described in `mydata.mrkmap`.

To run the program for the first time use:

```
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap --Ball true
```

To restrict the analysis to those markers overlapping each transcript try:

```
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap --Ball true --window 0
```

Additional covariates can be included from the cov.txt file. Try:

```
java -jar famCNV.jar --ped mydata.ped --trt mydata.trmap
        --ints mydata.int --mrk mydata.mrkmap --Ball true --cov cov.txt
```

Note that the computational time is strictly dependent on the hardware configuration, and for nuclear families such as those provided with the example files it is mostly determined by the CPU speed. The first example takes about 12 seconds to load the data and run the 45 tests on a MacBook Pro laptop with processor Intel Core 2 Duo 2.66 GHz. Therefore, using a similar hardware configuration, carrying out a genome-wide search using intensity data extracted for instance from a panel including 610K SNP probes would take less than 2 days of computational time.

When using larger families, famCNV could runs out of memory. This is because of the way that Java runs on a computer - what is actually run is a program called a virtual machine (the JVM) which executes the java instructions. The JVM has limits on the memory that can be allocated to the java program - and you might need to increase them if you are working with particularly large amount of data. In order to increase the amount of memory for famCNV, the program should be run from the command line by writing for example:

```
java -jar -Xms64M -Xmx256M famCNV.jar  --ped mydata.ped
        --trt mydata.trmap --ints mydata.int --mrk mydata.mrkmap
```

This sets the initial and maximum memory size to 64MB and 256MB. The M suffix can be changed with G to represent gigabyte.

Hint: To print a short help use the option `--help`.

8 Population stratification

famCNV uses information from both within- and between-family variance. Taking into account both types of variation leads to increased statistical power, but might be less robust to population stratification. Population stratification, for example due to ethnic admixture, can be a possible source of false positive associations. False positives could arise, for instance, if multiple subgroups within the sample differ both for the average trait value of quantitative traits and for the frequency of a CNV.

Several techniques have been developed to detect and correct for population stratification [1, 2, 3, 4]. If CNV association analysis is assessed using intensity measures from GWAS SNP data, SNP genotypes might be used for the identification of population structures in the sample.

For other sources of intensity data, such as those obtained through array comparative genomic hybridization (aCGH), a method such as Genomic Control [2] can be used to estimate inflation in the test statistics and apply a correction to the P values.

References

[1] Pritchard, J. K. & Rosenberg, N. A. Use of unlinked genetic markers to detect population stratification in association studies. 65, 220-228 (1999).

[2] Devlin, B. & Roeder, K. Genomic control for association studies. Biometrics 55, 997-1004 (1999).

[3] Pritchard, J. K., Stephens, M., Rosenberg, N. A. & Donnelly, P. Association mapping in structured populations. Am. J. Hum. Genet. 67, 170-181 (2000).

[4] Freedman, M. L. et al. Assessing the impact of population stratification on genetic association studies. Nature Genet. 36, 388-393 (2004).